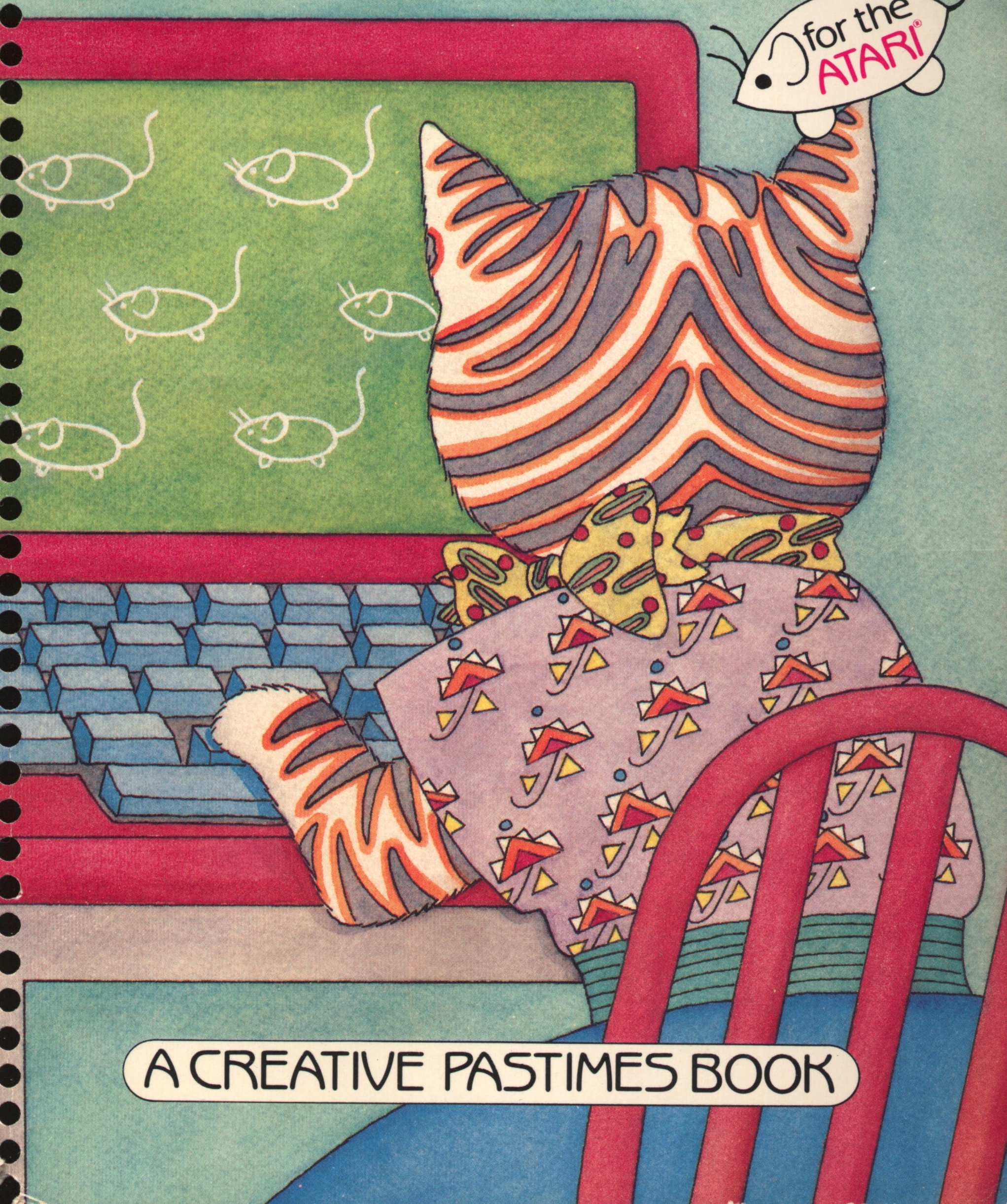
CLAIRE BAILEY PASSANTINO

MATILDA

THE COMPUTER CAT



A CREATIVE PASTIMES BOOK

Matilda the Computer Cat

for the ATARI® Computer

Claire Bailey Passantino

Text Illustrations by Nancy Gurganus

A Reston Computer Group Book
Reston Publishing Company, Inc.
A Prentice-Hall Company
Reston, Virginia

A CREATIVE PASTIMES BOOK

Library of Congress Cataloging in Publication Data

Passantino, Claire Bailey.
Matilda, the Computer Cat, for the ATARI computer.

(A Creative pastimes book)
"A Reston Computer Group book."
1. Atari 400 (Computer)—Programming. 2. Atari 800 (Computer)—Programming. 3. Basic (Computer program language) 4. Computer games. I. Title. II. Series.
QA76.8.A8P38 1984 001.64'25 83-26963
ISBN 0-8359-4323-2

This book is published by Reston Publishing Company, Inc., which is not affiliated with Atari, Inc., and Atari is not responsible for the accuracy or any representation made herein. ATARI is a registered trademark of Atari, Inc. 400 and 800 are trademarks of Atari, Inc.

Cover illustration by Bethann Thornburgh
Cover design by Carol Conway

10 9 8 7 6 5 4 3 2 1

Printed in the United States of America

Contents

Foreword

I hope you enjoy Matilda as much as I've enjoyed writing about her. She has worked her way into my heart as I've worked her into my computer. Maybe the same thing will happen to you.

This is the second of the Itty Bitty Bytes books for the ATARI® Computer. The programs were developed on the ATARI 400/800™ Computer. Each book in the series is designed to bring you twenty fun programs and, at the same time, teach you something about BASIC. Typing in the programs is hard work. You must be careful to number the lines exactly, spell the words correctly, and put in all the right punctuation. But you will feel extra good when your program runs smoothly!

Always read the explanations next to each program. Sometimes there are additions or changes that you can make to improve the program. Once you see what the program is all about, feel free to experiment with it—and by all means combine programs that work well together. The more you can do to make a program reflect your own special personality, the more you will enjoy it. Let me know if you find some nifty changes!

The Itty Bitty Bytes books have grown out of the teaching experiences I've had with my computer students. Special thank-yous go to all my "COMPU-KIDS"—and to my own kids, too—who keep bringing me one good idea after another. They are my severest critics, but my most outstanding assets.

Happy computing!

Claire Bailey Passantino

ATARI® is a registered trademark of Atari, Inc. 400 and 800 are trademarks of Atari, Inc.

A Note to Parents and Teachers

You bought the computer. You read the manuals. You did the demo programs. You and/or your children may even have taken some computer classes. But now the computer is just sitting there. Everyone was so enthusiastic. *What happened?*

The novelty of a new computer will take you just so far. Beyond that point, a continuing interest in this incredible tool is directly related to its usefulness. "Not useful" equates to "not used." So the problem becomes, what can computers do that children would find useful? What kinds of things would encourage children to expend the energy needed to create their own computer programs?

Each Itty Bitty Bytes book in the Creative Pastimes series is packed with computer activities appealing to young programmers. Simple games, contests, races, pictures, designs, songs, riddles, charts, tests, and more—all are designed to be fun while reinforcing beginning computer concepts and skills.

Besides providing fun, there are fringe benefits to having children write their own programs. In working with children, I have found that computer programming encourages them to:

- Think creatively.
- Use logical thinking skills.
- Attend to details.
- Take small steps to achieve a goal.
- Personalize programs.
- Develop pride and self-esteem.
- Appreciate packaged software.

To help children enjoy doing their own programs, here are some suggested DOs and DON'Ts:

- DO encourage children to type in their own programs. With younger children, bargain: "You type this line and I'll type the next one." (Save the long lines for yourself!)
- DON'T criticize typing expertise. Speed and correct fingering are typing skills that are minimally related to computer programming.
- DO allow children to make mistakes.
- DO help them find the errors they've made. (This is called "debugging" the program.)
- DON'T worry when there is an error message. This means that a mistake has been made. Check the program for "bugs." (The ATARI Computer will give you an "ERROR" when you go over a "READY" signal. Ignore this message. The computer is trying to "READ Y" and there is no "Y" to read.)

- DO encourage children to read and understand the program explanations.
- DON'T, however, force the issue. Some people learn by reading. Others learn by doing. As skills are repeated over and over in different contexts, children may just "catch on."
- DO be aware of some common pitfalls. Remember to:
  1. Use line numbers.
  2. Press RETURN after you type in a line.
  3. Give great attention to spelling and punctuation. Quotation marks, commas, semicolons, colons, and even spaces are very important.
  4. Save your program before you turn off the computer.
- DO help children save their programs on tape or disk so they can use them again and again. If you have a printer, use it to make "hard" copies of each program. People like to see themselves in print.
- DON'T enter commands in reversed or lower-case characters. (Reversed characters are produced by pressing "ᐟ|ᐠ." Press ᐟ|ᐠ again to return to regular characters. Lower-case letters are produced by pressing the "CAPS/LOWR" key. To return to capital letters, press the "SHIFT" and the "CAPS/LOWR" keys simultaneously.
- DO praise children for a job well done. And enjoy the programs that they've created.
- DO modify and use programs that you yourself find useful.

It is my sincere hope that the Itty Bitty Bytes books will help you and your children establish a healthy working relationship with your computer. Take that computer off the shelf! And let me know how things are working out!

Matilda the Computer Cat
for the ATARI® Computer

Main idea: Making an outline drawing using PLOT and DRAWTO

This is a simple drawing of Matilda. To get a better idea of how the program works, type in one section at a time: the head (lines 10 to 30), the whiskers (lines 39 to 70), the eyes (lines 79 to 80), the mouth (lines 89 to 100), and the bow (lines 110 to 140).

REM lines can be omitted, but they're nice to include to help the programmer remember what different sections of the program are all about.

Unless the colors are changed, they are as follows:

Color 1—Gold
Color 2—Light green
Color 3—Blue
Color 0—Black

To change the colors, you use a setcolor statement:

To change color 1, use SE.0,___,___
To change color 2, use SE.1,___,___
To change color 3, use SE.2,___,___
(the text window)
To change color 0, use SE.4,___,___
(the background)

Select a color (0 to 15) from the Setcolor Chart in the Appendix.

Select a shade, dark (0) to light (15).

To explain the criticism (lines 199–230):

200 It is necessary to allow a certain number of letters for an alphabetic variable before using it in the program. The A$ has to be "dimensioned" if we want to use it in line 220. We are allowing 50 bytes (letters and spaces) for the A$.
210 Prints the question in the text window.
220 A chance to answer the question.
230 No matter what you say, this message appears.

Meet Matilda

Type in the program
and you can meet Matilda!

```
10 GR.7
15 C.1
19 REM OUTLINE OF HEAD
20 PL.32,48:DR.54,16:DR.66,32:DR.72,32:DR.82,16:
   DR.104,48:DR.86,62
30 DR.54,62:DR.32,48
39 REM WHISKERS
40 PL.60,48:DR.32,52
50 PL.24,48:DR.60,48:DR.32,66
60 PL.84,46:DR.114,30
70 PL.120,46:DR.84,46:DR.112,62
79 REM EYES
80 PL.60,38:PL.80,38
89 REM MOUTH
90 PL.70,46:DR.70,56:DR.77,56:DR.63,56
100 PL.62,55:PL.78,55
110 REM PINK BOW
120 C.2:SE.1,5,10
130 PL.50,62:DR.88,62:DR.84,74:DR.70,62:DR.54,74:
    DR.50,62
140 PRINT "     HI THERE. I'M MATILDA!"
```

If you don't like my drawing, tell me so:

```
199 REM CRITICISM
200 DIM A$(50)
210 PRINT "WHAT DO YOU THINK OF THIS DRAWING?"
220 INPUT A$
230 PRINT "NOW DON'T BE CATTY!"
```

Main idea: A vertical race up the screen

5 Allows 20 spaces for the N$ (used in line 150).

10 Graphics mode 5.

20 The background becomes blue (like the sky).

30 The text window becomes green (like the grass).

35 Color 1 becomes black (for the trees).

40–60 Draws the trees.

70 Prints the cats' names in the text window.

80 H stands for Hairy. M stands for Macho. HS stands for Hotshot. All 3 cats will start at box 38 (at the bottom of the screen).

85–95 Draws a spot on the tree and a spot of sky underneath. (Each time a spot is drawn, there is a spot of sky just below it. This erases the old spot, so it looks as though the new spot has moved up the tree!)

96 If any spot—I mean cat—gets to the top of the screen, he has won the race. Skips to line 150.

100 Picks a number from 1 to 3.

110 If a 1 is picked, Hairy moves up the tree.

120 If a 2 is picked, Macho moves up the tree.

130 If a 3 is picked, Hotshot moves up the tree.

145 Goes to line 85 to plot a new spot on one of the trees.

150 You look at the drawing and tell who won.

160 Prints a message.

Change line 70 if you want to rename these characters!

The Dating Game

Three cool cats are vying for Matilda's attention. She tells them that she will date the best tree climber. Who will win?

```
5 DIM N$(2Ø)
1Ø GR.5
2Ø SE.4,8,8
3Ø SE.2,12,4
35 C.1:SE.Ø,Ø,1
4Ø FOR X=15 TO 55 STEP 2Ø
5Ø PL.X,39:DR.X,Ø
6Ø NEXT X
7Ø PRINT "HAIRY","MACHO","HOTSHOT"
8Ø H=38:M=38:HS=38
85 C.2
9Ø PL.14,H:PL.34,M:PL.54,HS
95 C.Ø:PL.14,H+1:PL.34,M+1:PL.54,HS+1
96 IF H=Ø OR M=Ø OR HS=Ø THEN PRINT "WHO WON?":GOTO
   15Ø
1ØØ X=INT(3*RND(1))+1
11Ø IF X=1 THEN H=H-1
12Ø IF X=2 THEN M=M-1
13Ø IF X=3 THEN HS=HS-1
145 GOTO 85
15Ø INPUT N$
16Ø PRINT N$;" GETS A DATE WITH MATILDA!!"
```

Main idea: Using READ . . . DATA to play a song

10 Graphics mode 2 is a text mode with big letters.

20 Positions the message 3 blocks over and 4 blocks down from the upper left corner of the screen.

30 The computer will read 2 pieces of data, starting with line 100. N stands for the note. T stands for the time (number of beats the note gets).

31 When the computer reads the number "−1" for a note, it skips to line 200. (Look at line 140. This is called "flag data." Flag data acts as a signal for the computer to do something special.)

40 Plays the note: SOUND 0 (the first "voice"), N (the note), 10 (a clear tone), 8 (at medium loudness).

50 Regulates how fast the song will be played. The computer takes the beats a note will play, say 3, and holds the note for 80*3, or 240 units of computer time. To have the song play faster, change the 80 to a smaller number. To have the song play slower, change the 80 to a larger number.

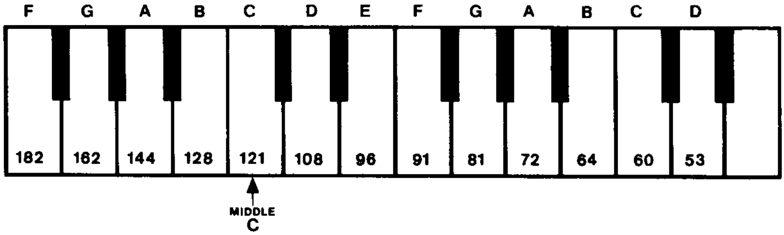
55 Turns off the sound.

60 Goes to line 30 to read the next note and the number of beats.

100–130 The data for the song (see explanation below).

140 The flag data.

200 An ending message.



Where do the numbers for the notes come from? The computer has sounds from 0 to 255. Numbers can be assigned to the notes in the scale. This song starts with E–D–C (hence 96–108–121). E plays for 3 beats (hence 96,3). D plays for 3 beats (hence 108,3). C plays for 6 beats (hence 121,6).

If you're really ambitious, you could make up another computer song! For more notes, see the Appendix (page 53).

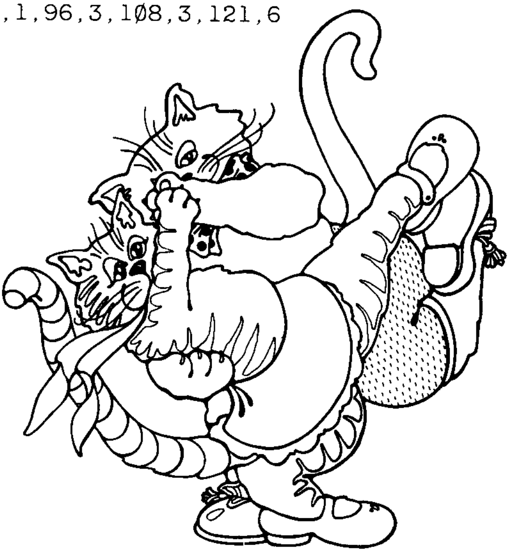
Our Song

Matilda decides she likes dancers better than tree climbers. She meets a really good dancer named Tom. Every night they go to the neighborhood disco and dance to their favorite song.

```
10 GR.2
20 POS.3,4:PRINT#6;"OUR SONG !!!"
30 READ N,T
31 IF N=-1 THEN 200
40 SO.0,N,10,8
50 FOR P=1 TO 80*T:NEXT P
55 SO.0,0,0,0
60 GOTO 30
100 DATA 96,3,108,3,121,6
110 DATA 96,3,108,3,121,6
120 DATA 81,3,91,2,91,1,96,6
130 DATA 81,3,91,2,91,1,96,6
140 DATA -1,-1
200 PRINT "ALL THE CATS LOVE THIS ONE!"
```

Want to hear the rest of the song?

```
131 DATA 81,1,60,2,60,1,64,1,72,1,64,1,60,2,81,1,81,2
132 DATA 81,1,60,2,60,1,64,1,72,1,64,1,60,2,81,1,81,2
133 DATA 81,1,60,2,60,1,64,1,72,1,64,1,60,2,81,1,81,2
134 DATA 91,1,96,3,108,3,121,6
```



Main idea: Initializing and incrementing a variable

10 The counter for LM (loves me) is set to 0. The counter for LN (loves me not) is set to 0.

20–70 Everything in this loop will be done 365 times, since there are 365 days in the year:

30 The computer picks a number, a 0 or a 1.

40 If it picked a 0, LM (loves me) goes up 1 point.

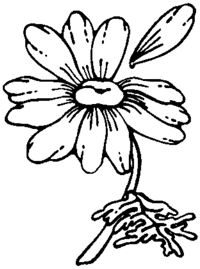
50 If it picked a 1, LN (loves me not) goes up 1 point.

60 Prints the current tally.

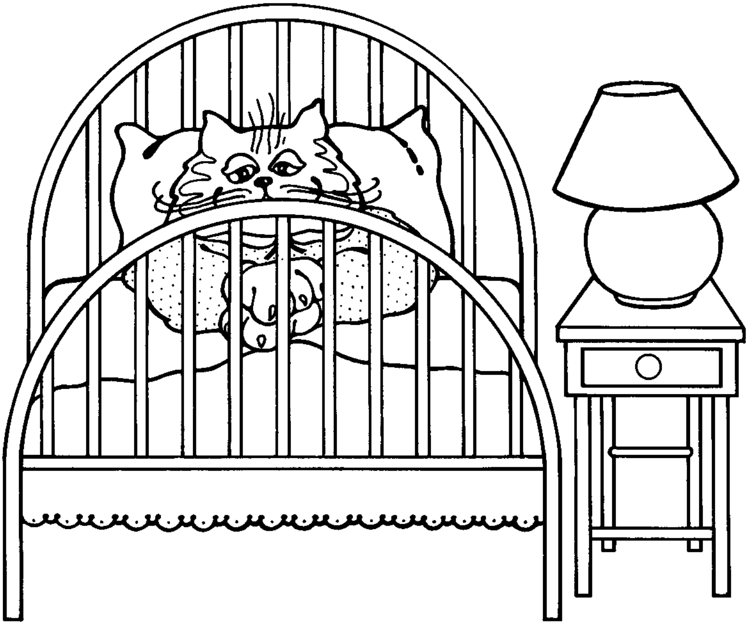
80 If LM is over half of 365, Matilda decides that Tom loves her.

90 If the LM score is under half of 365, she decides that he hates her.

A year is a long time to take for a decision like this! To make the time shorter, change the number in line 20. Change lines 80 and 90 accordingly.



Does He Love Me?



Every night Matilda asks herself, "Does he love me?" After one year, this is what she decides.

```
1Ø LM=Ø:LN=Ø
2Ø FOR Q=1 TO 365
3Ø X=INT(2*RND(1))
4Ø IF X=Ø THEN LM=LM+1
5Ø IF X=1 THEN LN=LN+1
6Ø PRINT"LOVES ME ";LM,"LOVES ME NOT ";LN
7Ø NEXT Q
8Ø IF LM>=183 THEN PRINT "HE LOVES ME! EXCELLENT!"
9Ø IF LM<183 THEN PRINT "HE HATES ME. RATS!"
```

Main idea: Using OPEN . . . GET for instant keyboard input

10 Graphics 2 gives you big letters and the +16 removes the text window.
15 Changes Color 1 to red, so you get red hearts.
20 Gives you lower-case letters in Graphics 2.
25–50 Blackens out a rectangle in the middle of the screen.
60 Positions and prints "matilda." Press the /|\ key to get reversed characters. This prints "matilda" in blue.
70 Positions and prints "loves" in Color 1 (which became red in line 15).
80 Positions and prints "tom" in blue.
100 Goes to itself over and over again—an infinite loop. This freezes the image on the screen. Leave it out and see what happens.

The additional section allows you to make some startling changes.

5 Allows you to press a key for an "instant input."
70 Positions the cursor to get ready to print something.
75 The ASCII code for H is 72. If you press H, this line directs the computer to print "hates" in blue.
76 The ASCII code for L is 76. If your press L, this line directs the computer to print "loves" in red.
100 Tells the computer to get the ASCII code number of the key you pressed.
110 If you pressed H, color 1 becomes black and the hearts disappear (because they are black on a black background), and you go back into an infinite loop at line 20.
120 If you pressed L, color 1 becomes red and the hearts appear (because they are red on a black background), and you go back into an infinite loop at line 20.

(Now that you have lines 110 and 120, you don't need the infinite loop in the original line 100 above.)

Is It Love?

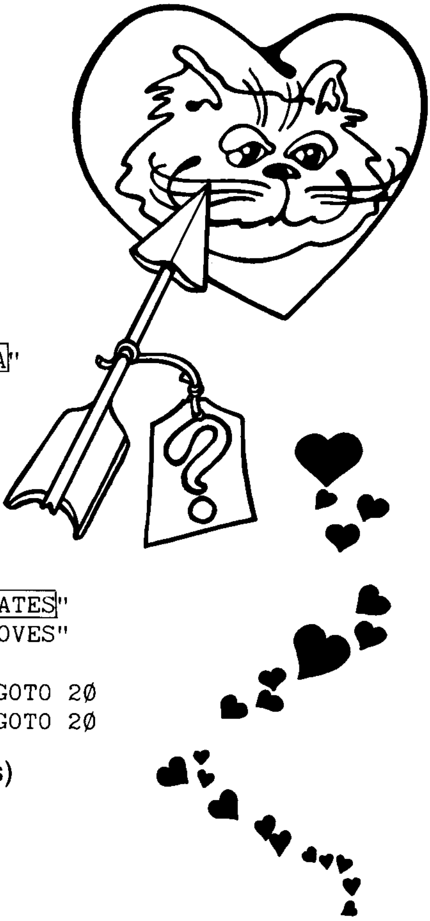
Matilda loves Tom.

```
1Ø GR.2+16
15 SE.Ø,4,4
2Ø POKE 756,226
25 C.2
26 SE.1,Ø,Ø
3Ø FOR Y=3 TO 7
4Ø PL.3,Y:DR.16,Y
5Ø NEXT Y
6Ø POS.6,4:PRINT#6;"MATILDA"
7Ø POS.7,5:PRINT#6;"LOVES"
8Ø POS.8,6:PRINT#6;"TOM"
1ØØ GOTO 1ØØ
```

Or does she?

```
5 OPEN #1,4,Ø,"K"
7Ø POS.7,5
75 IF N=72 THEN PRINT#6;"HATES"
76 IF N=76 THEN PRINT#6;"LOVES"
1ØØ GET #1,N
11Ø IF N=72 THEN SE.Ø,Ø,Ø:GOTO 2Ø
12Ø IF N=76 THEN SE.Ø,4,4:GOTO 2Ø
```

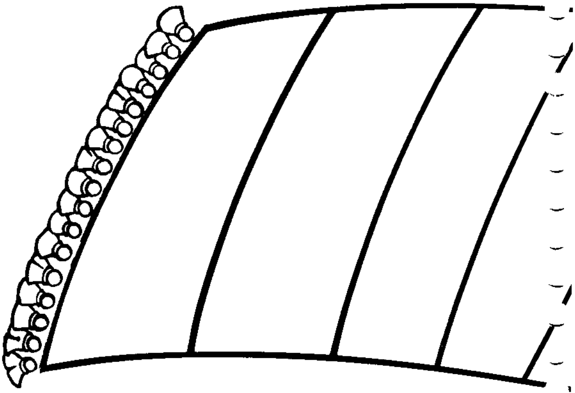
(Press L for loves, H for hates)



Main idea: Using control characters to make a design

When you press the control key (CTRL), you can make some pretty designs if you press another key at the same time. This program uses CTRL–F and CTRL–G to make a herringbone pattern. For a different design, simply change lines 40 and 50.

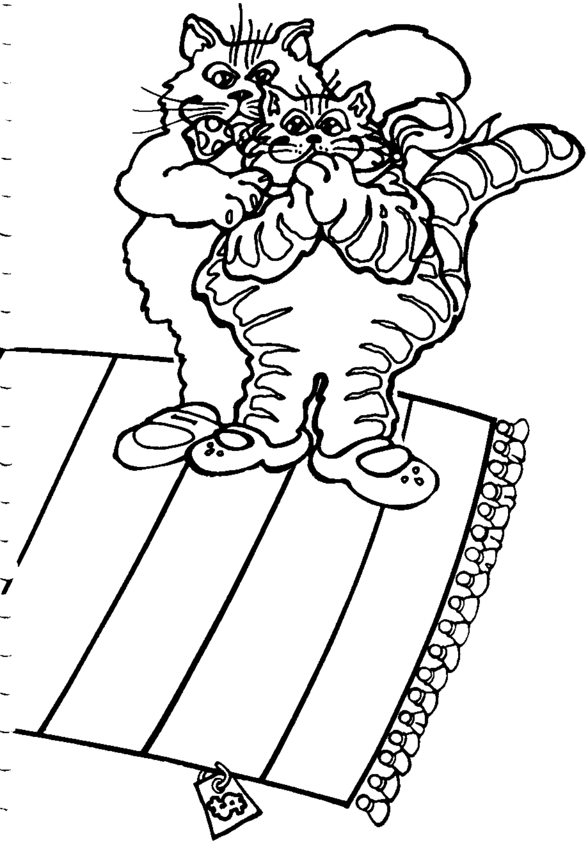
10 Graphics 0 clears the screen.
20 Skips 2 lines.
30–60 The program will repeat this loop 10 times:
40 Prints "//////////////". (This row is accomplished by pressing CTRL and F at the same time.)
50 Prints "\\\\\\\\\\\\\\". (This row is accomplished by pressing CTRL and G at the same time.)
70 A dummy line freezes the image on the screen without the READY signal.



A New Rug!

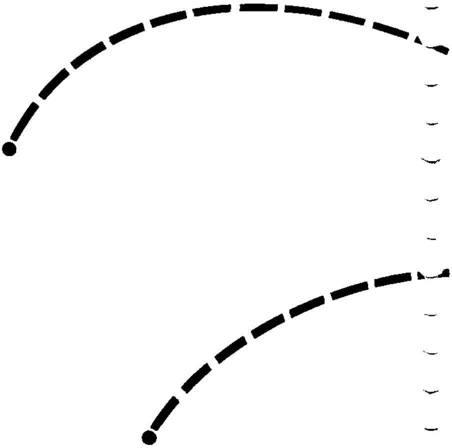
Tom and Matilda went to a big flea market. What a beautiful rug Matilda bought. And so cheap, too! (Cats love herringbone, you know!)

```
1Ø GR.Ø
2Ø PRINT:PRINT
3Ø FOR P=1 TO 1Ø
4Ø PRINT "///////////////////////////"
5Ø PRINT "\\\\\\\\\\\\\\\\\\\\\\\\\\\"
6Ø NEXT P
7Ø GOTO 7Ø
```



Main idea: Guessing game

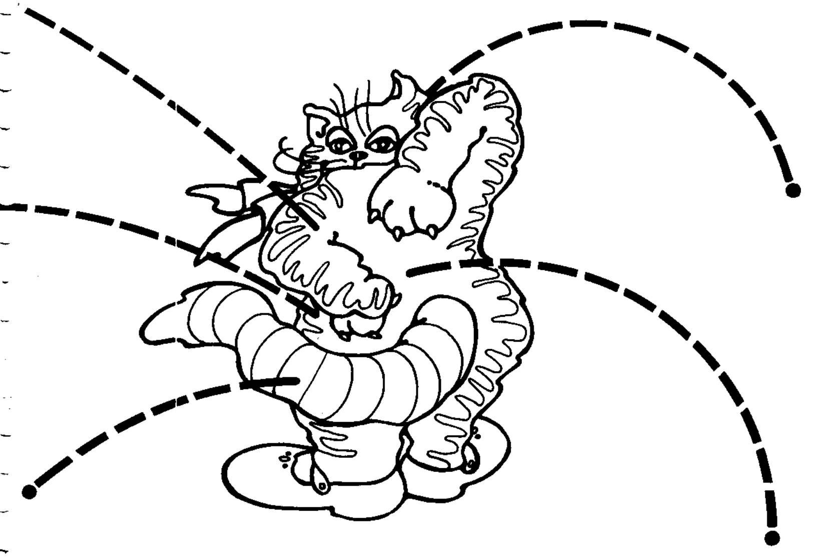
5 Graphics 0 clears the screen.
10–30 Prints the directions.
40 Picks a number from 0 to 199.
50 Allows you to input a guess.
60 If you guessed correctly, the computer goes to line 100.
70 If you guessed too low, it prints "MORE!"
80 If you guessed too high, it prints "FEWER!"
90 Goes to line 50 to allow you another guess.
100 Prints a message.
110–130 Counts the number of fleas.
120 You get the design by pressing CTRL and T at the same time.



Fleas!

Unfortunately, the rug had fleas. Poor Matilda! Guess how many fleas she counted just today!

```
5 GR.Ø
1Ø PRINT "MATILDA HAS A BAD CASE OF FLEAS!"
2Ø PRINT "GUESS HOW MANY SHE COUNTED TODAY!"
3Ø PRINT "HINT: LESS THAN 2ØØ!"
4Ø F=INT(2ØØ*RND(1))
5Ø INPUT G
6Ø IF G=F THEN 1ØØ
7Ø IF G<F THEN PRINT "MORE!"
8Ø IF G>F THEN PRINT "FEWER!"
9Ø GOTO 5Ø
1ØØ PRINT "HERE THEY COME"
11Ø FOR N=1 TO F
12Ø PRINT N;" ";"●",
13Ø NEXT N
```



Main ideas: Initializing, incrementing, and decrementing a variable

10 Graphics 0 clears the screen.

20 L stands for the number of lives Tom has. Tom starts the game with 9 lives. This is called *initializing a variable.*

30 Picks a number from 1 to 5 to select the girlfriend that Tom will visit.

40–50 Asks Matilda to guess where Tom is.

60 Allows you to input a guess.

70 If Tom *isn't* there, the program goes back to line 30 so the computer can choose the next girlfriend to get a visit.

80 If Tom *is* there, it prints a message and goes to line 90.

90 Subtracts 1 from the number of lives. This is called *decrementing.*

100 If Tom has any lives left, it tells how many and goes back to line 30 so the computer can pick the next girlfriend. (Tom never learns!)

110 When Tom runs out of lives, the screen is cleared with Graphics 2.

120 Positions and prints a farewell message to Tom.

How long did Tom get away with it?

25 Sets V (the number of visits) to 0. (Tom starts out with no visits to his girlfriends.)

70 This line is changed so that each time Tom gets away with it, you add 1 to the number of visits before going back to line 30. This is called *incrementing a variable.*

115–116 Jazzes up the screen. The background is light blue and the letters are dark blue. Blue is such a heavenly color!

130 Tells how many visits Tom got away with during his 9 lives. (He died smiling!)

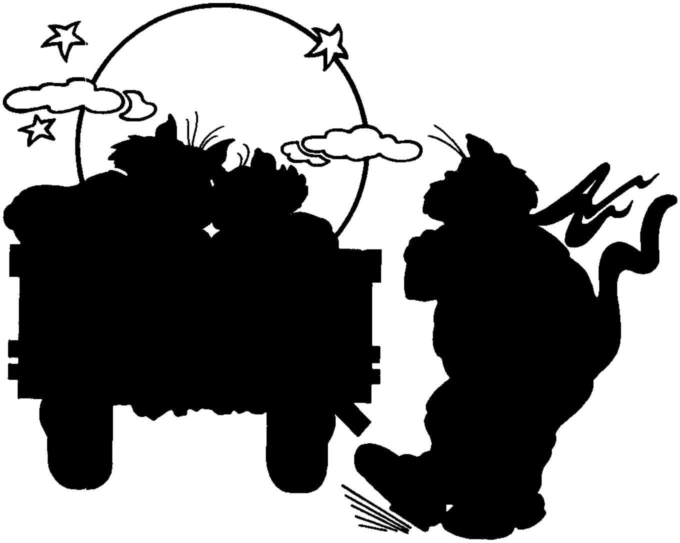
Nine Lives

Oh no! Tom has *five* other girlfriends! Everyday he visits one of them. If Matilda finds out, she will kill him. You may play this game till Tom loses all of his nine lives.

```
10 GR.0
20 L=9
30 T=INT(5*RND(1))+1
40 PRINT "OK,MATILDA"
50 PRINT "WHERE DO YOU THINK TOM IS TODAY?(1-5)"
60 INPUT G
70 IF T<>G THEN GOTO 30
80 IF T=G THEN GR.0:PRINT "YOU'RE DEAD, TOM!"
90 L=L-1
100 IF L>0 THEN PRINT L;" LIVES TO GO!":GOTO 30
110 GR.2
120 POS.3,3:PRINT#6;"BYE BYE TOM!"
```

How long did he get away with it?

```
25 V=0
70 IF T<>G THEN V=V+1:GOTO 30
115 SE.4,8,8
116 SE.0,8,10
130 PRINT "BUT YOU GOT AWAY WITH IT ";V;" TIMES!"
```



Main ideas: A horizontal race
Incrementing variables

10 N1$ stands for Name 1. N2$ stands for Name 2. N3$ stands for Name 3. This line allows 15 letters for each name.

11 All 3 names will begin at box 0.

15 Graphics 0 clears the screen.

20–40 Allows you to input your choices for the 3 names.

45 Clears the screen.

46 Removes the cursor.

50–70 Positions and prints the 3 names.

80 Picks a number from 1 to 3.

90–110 If the computer picked a 1, it goes to the line under Name 1 (the line under N1$) and prints a star. Then it adds 1 to N1, so the next time the star will be printed one box over. If a 2 is picked, Name 2 gets a star. If a 3 is picked, Name 3 gets a star.

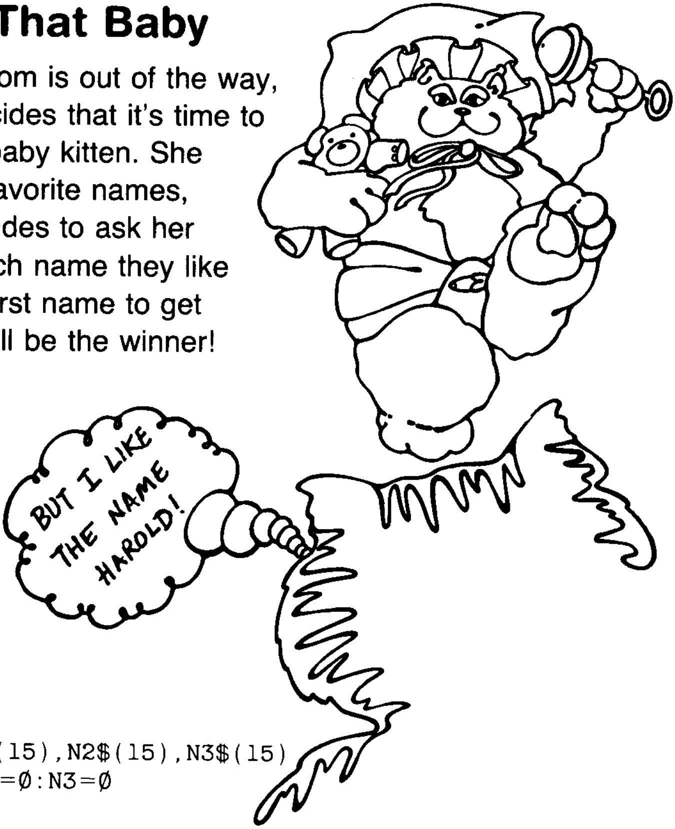
115 If any name gets to box 39 (which is the 40th box, counting box 0), then we have a winner. The computer goes to line 200.

120 If no name has won yet, the computer goes back to line 80 to pick another number.

200 A dummy line to prevent the READY signal.

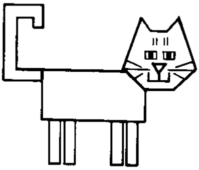
Name That Baby

Now that Tom is out of the way, Matilda decides that it's time to name her baby kitten. She has three favorite names, so she decides to ask her friends which name they like best. The first name to get 40 votes will be the winner!



```
1Ø DIM N1$(15),N2$(15),N3$(15)
11 N1=Ø:N2=Ø:N3=Ø
15 GR.Ø
2Ø PRINT "WHAT IS NAME 1";:INPUT N1$
3Ø PRINT "WHAT IS NAME 2";:INPUT N2$
4Ø PRINT "WHAT IS NAME 3";:INPUT N3$
45 GR.Ø
46 POKE 752,1
5Ø POS.1,5:PRINT N1$
6Ø POS.1,1Ø:PRINT N2$
7Ø POS.1,15:PRINT N3$
8Ø X=INT(3*RND(1))+1
9Ø IF X=1 THEN POS.N1,6:PRINT "*":N1=N1+1
1ØØ IF X=2 THEN POS.N2,11:PRINT "*":N2=N2+1
11Ø IF X=3 THEN POS.N3,16:PRINT "*":N3=N3+1
115 IF N1>39 OR N2>39 OR N3>39 THEN GOTO 2ØØ
12Ø GOTO 8Ø
2ØØ GOTO 2ØØ
```

Main ideas: Nested FOR . . . NEXT loops
Using the SELECT key



This program is an excellent illustration of something called "nested loops." When using FOR . . . NEXT loops, it is important to "nest" them properly or you will get an error message. Compare the following loops.

Properly nested loop

```
FOR X=1 TO 10
FOR Y=1 TO 5
PLOT X,Y
NEXT Y
NEXT X
```

Crossed loop

```
FOR X=1 TO 10
FOR Y=1 TO 5
PLOT X,Y
NEXT X
NEXT Y
```

There are 10 nested loops in this program. Can you find them?

Let's see how the program works:

10 The drawing will be done in Graphics mode 3.
15 Color 1 is gold. It will be changed later in line 230.
19–60 Draws the body of the pajamas from X=9 to X=29, and from Y=4 to Y=14.
69–200 Draws 4 sleeves on the pajamas. Each sleeve is 3 boxes wide and extends from vertical point 15 to 18.
210–280 This section is really a color demo. When the SELECT key is pressed, the program skips to line 300.

For stripes and checks, experiment with lines 20 and 30.

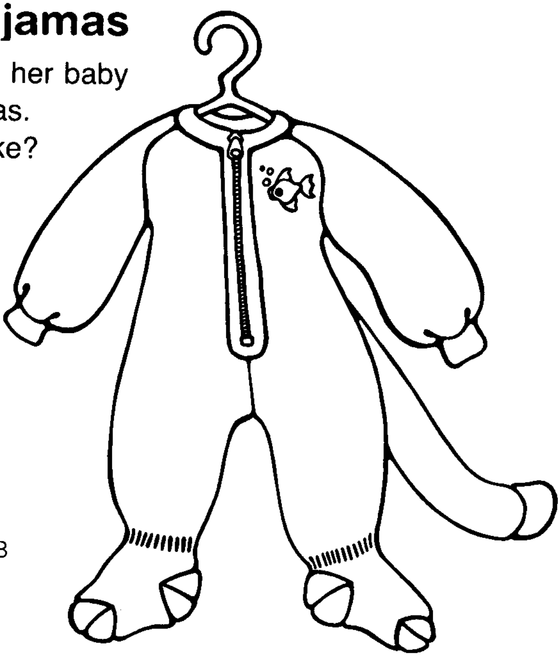
```
2Ø FOR X=9 TO 29 STEP 2
3Ø FOR Y=4 TO 14 STEP 2
```

And here's a nifty addition from my daughter Kristin:

```
4 DIM A$(2)
5 G=3
1Ø GR.G
31Ø PRINT "WOULD YOU LIKE A SMALLER SIZE? (Y,N)"
32Ø INPUT A$
33Ø IF A$="N" THEN END
34Ø G=G+2
35Ø IF G>7 THEN PRINT "SORRY, KITTIES DON'T COME THAT
    SMALL":GOTO 35Ø
36Ø GOTO 1Ø
```

The Cat's Pajamas

Matilda wants to buy her baby a new pair of pajamas. Which pair do you like?



```
10 GR.3
15 C.1
19 REM BODY
20 FOR X=9 TO 29
30 FOR Y=4 TO 14
40 PL.X,Y
50 NEXT Y
60 NEXT X
69 REM SLEEVES
70 FOR L1=10 TO 12
80 FOR L2=14 TO 16
90 FOR L3=22 TO 24
100 FOR L4=26 TO 28
110 FOR V=15 TO 18
120 PL.L1,V
130 PL.L2,V
140 PL.L3,V
150 PL.L4,V
160 NEXT V
170 NEXT L4
180 NEXT L3
190 NEXT L2
200 NEXT L1
```

Press SELECT to select the pajamas you like the best.

```
210 FOR C=0 TO 15
220 FOR SH=0 TO 15
230 SE.0,C,SH
240 FOR P=1 TO 50:NEXT P
250 A=PEEK(53279)
260 IF A=5 THEN GOTO 300
270 NEXT SH
280 NEXT C
300 PRINT "          THE CAT'S PAJAMAS"
```

Main idea: Inserting variables into a text

4 We are allowing 15 letters for each input variable.
5 Graphics 0 clears the screen.
10–75 Asks for input.
100 A remark.
105 Skips 2 lines.
110–160 Inserts inputs into the poem.

As you type, notice the use of spaces and semicolons. The semicolon acts as a formatting direction to the computer. It tells the computer: The first thing to be printed after the semicolon will appear *right next to* the last thing printed before the semicolon. For example, look at line 120:

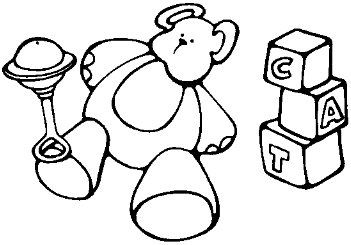
```
120 PRINT "THE CAT AND THE " ;I$
                          ↖space
```

If you put in the word PIANO for `I$`, your printout would read:

```
THE CAT AND THE PIANO
```

Without the space, it would read:

```
THE CAT AND THEPIANO
```



Hey Diddle Diddle

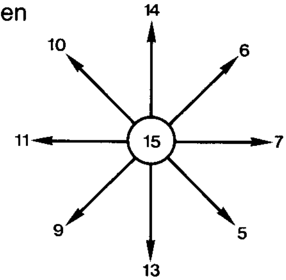
Matilda is teaching nursery rhymes to Harold. Sometimes she forgets the words. Maybe you can help her out!

```
4 DIM I$(15),AN$(15),SK$(15),
  AN2$(15),TH$(15),TH2$(15)
5 GR.Ø
6 PRINT "PLEASE GIVE ME:"
1Ø PRINT "A MUSICAL INSTRUMENT";
15 INPUT I$
2Ø PRINT "AN ANIMAL";
25 INPUT AN$
3Ø PRINT "SOMETHING IN THE SKY";
35 INPUT SK$
4Ø PRINT "ANOTHER ANIMAL";
45 INPUT AN2$
5Ø PRINT "TWO THINGS ON THE DINNER TABLE"
6Ø PRINT "1.";
65 INPUT TH$
7Ø PRINT "2.";
75 INPUT TH2$
1ØØ REM PRINT POEM
1Ø5 PRINT:PRINT
11Ø PRINT "HEY DIDDLE DIDDLE"
12Ø PRINT "THE CAT AND THE ";I$
13Ø PRINT "THE ";AN$;" JUMPED OVER THE ";SK$
14Ø PRINT "THE LITTLE ";AN2$;" LAUGHED"
15Ø PRINT "TO SEE SUCH SPORT"
16Ø PRINT "AND THE ";TH$;" RAN AWAY WITH THE ";TH2$
```

Main idea: Using the joystick to draw on the screen

Before you start, type in this little program:

```
1 PRINT STICK(0)
2 GOTO 1
```



Push the joystick in all directions.
Can you get all the numbers?

Now type NEW and start the program!

4 Since we have 3 kittens, we are going through the routine 3 times.
5 Graphics mode 3 will be used.
6 Color 1 is gold.
10 Prints a message in the text window.
20–21 Selects a number (M) from 0 to 39 and a number (N) from 0 to 19. The point (M,N) is where the mittens are hidden!
29 The starting point of your search is in the lower right corner.
30 Plots the point where you are.
40 ST stands for the position of the first joystick, STICK(0). (Make sure it is plugged into the first jack!)
50–80 Depending on which direction you push, you will change the value of X, the value of Y, *or* both values. For example, if you push →, STICK(0) will be 7 and you will add 1 to the X only (X=X+1). And if you push ↓, STICK(0) will be 13, and you will add 1 to Y only (Y=Y+1). But if you push ↘, STICK(0) will be 5 and you will add 1 to both X *and* Y (X=X+1 and Y=Y+1).
90–120 Keeps you from getting an error message by keeping the point on the screen.
130 Checks to see if you are within 2 units of the selected point (M,N). To make the game easier, change the 2 to a 3 or 4!
140 Goes back to line 130 to plot the point where you are now.
200 Prints a congratulatory message.
210 Pauses.
220 Goes back to line 4 for another giant loop.

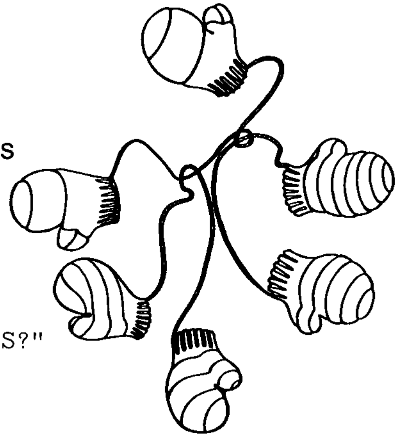
For a flashy ending each time you find a pair of mittens:

205 Goes through the loop 50 times.
206 Selects a random setcolor from 0 to 15.
207 Changes SE.4 (the background color).
208 Next flash.

Lines 230 to 260 provide a final message when you've found all the missing mittens.

Lost Mittens

Harold and his two friends
are always losing their mittens.
Please help the three little kittens
and *find those mittens!*



```
4 FOR T=1 TO 3
5 GR.3
6 C.1
1Ø PRINT "WHERE ARE MY MITTENS?"
2Ø M=INT(4Ø*RND(1))
21 N=INT(2Ø*RND(1))
29 X=39:Y=19
3Ø PL.X,Y
4Ø ST=STICK(Ø)
5Ø IF ST=6 OR ST=7 OR ST=5 THEN X=X+1
6Ø IF ST=9 OR ST=11 OR ST=1Ø THEN X=X-1
7Ø IF ST=1Ø OR ST=14 OR ST=6 THEN Y=Y-1
8Ø IF ST=9 OR ST=13 OR ST=5 THEN Y=Y+1
9Ø IF X>39 THEN X=39
1ØØ IF X<Ø THEN X=Ø
11Ø IF Y>19 THEN Y=19
12Ø IF Y<Ø THEN Y=Ø
13Ø IF X<M+2 AND X>M-2 AND Y<N+2 AND Y>N-2 THEN 2ØØ
14Ø GOTO 3Ø
2ØØ PRINT "YOU FOUND THEM!"
21Ø FOR P=1 TO 5ØØ:NEXT P
22Ø NEXT T
```

You deserve a reward:

```
2Ø5 FOR F=1 TO 5Ø
2Ø6 C=INT(16*RND(1))
2Ø7 SE.4,C,8
2Ø8 NEXT F
```

And here's a big thank-you:

```
23Ø GR.2
24Ø POS.1,3:PRINT#6;"3 CHEERS FOR YOU!"
25Ø POS.3,5:PRINT#6;"MEW MEW MEW!!!"
26Ø GOTO 26Ø
```

Main idea: Designing a simple test

This is an example of one kind of test you can make up on the computer.

5 C$ (the "cat" word) is allowed 15 letters. D$ (the description of the "cat" word) is allowed 30 letters. G$ (the guess at the "cat" word) is allowed 15 letters.

10 Graphics 2 is used for big letters.

15–20 Positions and prints the title of the test.

30 Reads the "cat" word and its description. (The data begins at line 300.)

35 Checks for the flag data (see line 390).

40 Prints the description of the "cat" word.

50 Waits for the test-taker to input a guess.

55 The position for the "RIGHT" or "WRONG" message to follow.

60 If you guessed the correct "cat" word, then it prints "RIGHT." Spelling counts, and don't put in any extra spaces!

70 If you guessed wrong, it prints "WRONG" and tells you the right answer.

75 Pauses.

80 Goes back to line 10 to clear the screen and start over with the next word.

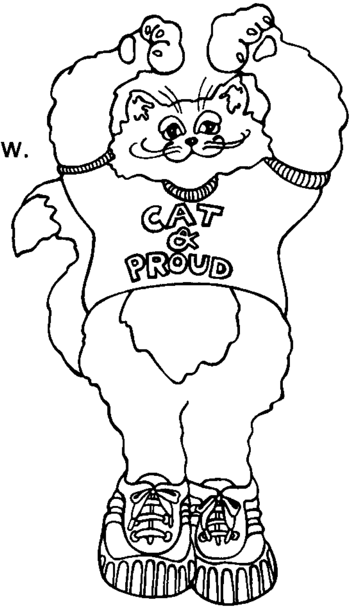
300–370 The data lines are arranged as C$,D$ to match line 30.

390 The flag data—a signal to the computer to do something. In line 35, when the computer reads this flag data, it prints "CAT'S ALL FOLKS!" and ends.

Do you know other "cat" words? You may add them to the test using lines 371 to 389. Do not remove line 390. You need it to end the program! Without it, you will get an error message when the computer runs out of data!

Cat Heritage

Matilda wants her baby to grow up proud to be a cat. Every day as he gets older, Matilda teaches him the words that every cat should know. To check his vocabulary, sometimes she gives him this little test.



```
5 DIM C$(15),D$(3Ø),G$(15)
1Ø GR.2
15 POS.4,3
2Ø PRINT#6;"THINK CATS!"
3Ø READ C$,D$
35 IF C$="END" THEN POS.2,5:
   PRINT#6;"CAT'S ALL
   FOLKS!":END
4Ø PRINT D$
5Ø INPUT G$
55 POS.6,5
6Ø IF G$=C$ THEN PRINT#6;"RIGHT"
7Ø IF G$<>C$ THEN PRINT#6;"WRONG":PRINT "IT WAS ";C$
75 FOR P=1 TO 6ØØ:NEXT P
8Ø GOTO 1Ø
3ØØ DATA CATERPILLAR,A BABY BUTTERFLY
31Ø DATA CATNIP,A CAT'S FAVORITE PLANT
32Ø DATA CATCHER,A BASEBALL PLAYER
33Ø DATA CATALOG,WHERE CATS BUY THEIR CLOTHES
34Ø DATA CATTLE,A HERD OF CATS
35Ø DATA CATSUP,KETCHUP FOR KITTIES
36Ø DATA CATASTROPHE,A TRIP TO THE VET
37Ø DATA CATCHOO,WHEN CATS SNEEZE
39Ø DATA END,END
```

Here's a flashy reward for a right answer:

```
6Ø IF G$=C$ THEN PRINT#6;"RIGHT":GOSUB 2ØØ
2ØØ REM FLASH
21Ø FOR F=1 TO 3Ø
22Ø C=INT(16*RND(1))
23Ø SE.4,C,6
24Ø NEXT F
25Ø RETURN
```

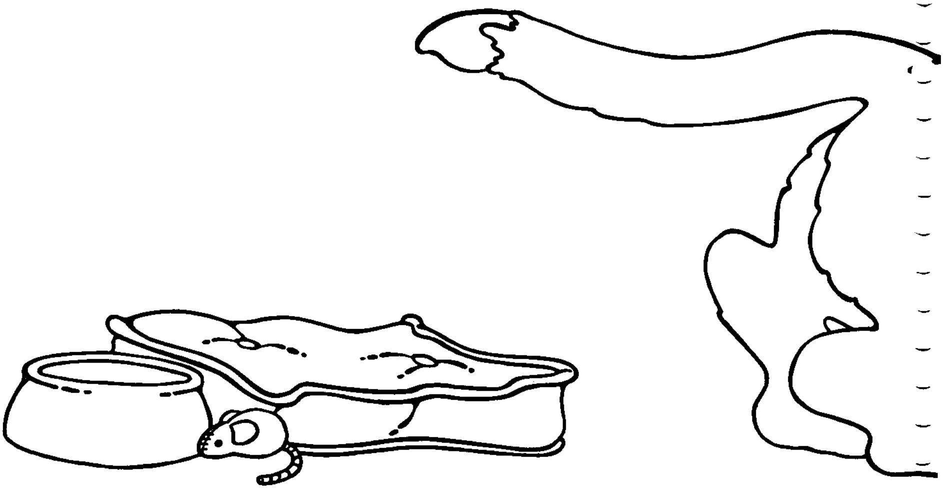
Main ideas: Making a chart
Using conversion formulas

10 Graphics 0 clears the screen.
20 Asks for the cost of a can of cat food.
30 Allows for your input. (Enter the cost as a decimal. For example, enter .39 for 39 cents.)
40 Asks how many cans are used in a day.
45 Allows for your input.
50 Clears the screen.
60–90 Positions and prints the title of the chart.
100 Skips a line.
105 Figures out the cost for a day. D stands for the daily cost. C stands for the cost of a can. N stands for the number of cans. D=C*N.
110–140 Figures and prints costs on the chart.

Notice the comma after each PRINT command. For example, look at line 110:

```
11Ø PRINT,"DAILY","$";D
         ↖comma
```

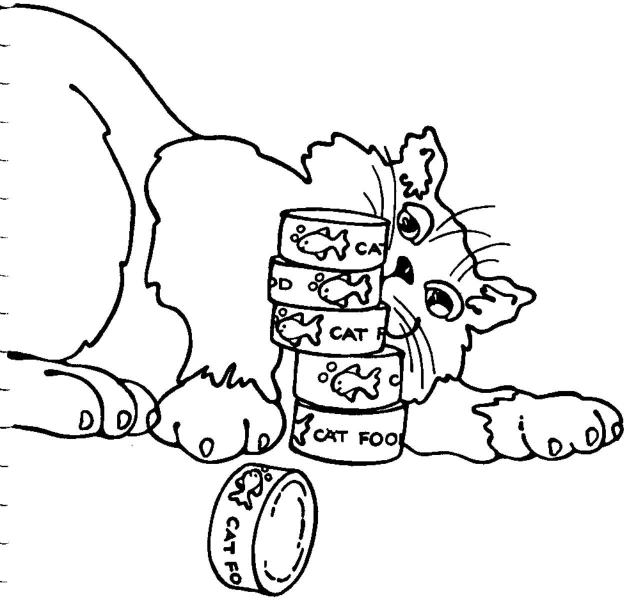
The comma after the word PRINT moves the cursor to the second print zone before printing.



Cost of Living

Harold has developed quite an appetite. Matilda is trying to figure out how much money she spends on cat food. You can do this, too. Just type in this program!

```
1Ø GR.Ø
2Ø PRINT "HOW MUCH DOES ONE CAN COST?"
3Ø INPUT C
4Ø PRINT "HOW MANY CANS DO I USE IN ONE DAY?"
45 INPUT N
5Ø GR.Ø
6Ø POS.12,5
7Ø PRINT "COST OF CAT FOOD"
8Ø POS.12,6
9Ø PRINT "----------------"
1ØØ PRINT
1Ø5 D=C*N
11Ø PRINT,"DAILY","$";D
12Ø PRINT,"WEEKLY","$";7*D
13Ø PRINT,"MONTHLY","$";3Ø*D
14Ø PRINT,"YEARLY","$";365*D
```



Main ideas: Random drawing on the screen
Using the OPTION key

10 Graphics 7. The +16 removes the text window.
15 Color 1 is gold.
20 The upper left corner is the starting point.
25–80 This loop goes through 50 repetitions:
30 Picks an X number from 0 to 159.
40 Picks a Y number from 0 to 95.
50 Draws to the point picked.
60 Picks a sound (S) from 0 to 255.
70 Plays the sound.
80 End of the loop.
100 Goes back to line 10 where the screen is cleared and you start over. This program is in an infinite loop!

Changing colors:

16 Picks a C number from 0 to 15.
17–18 Sets the color of the drawing and background.

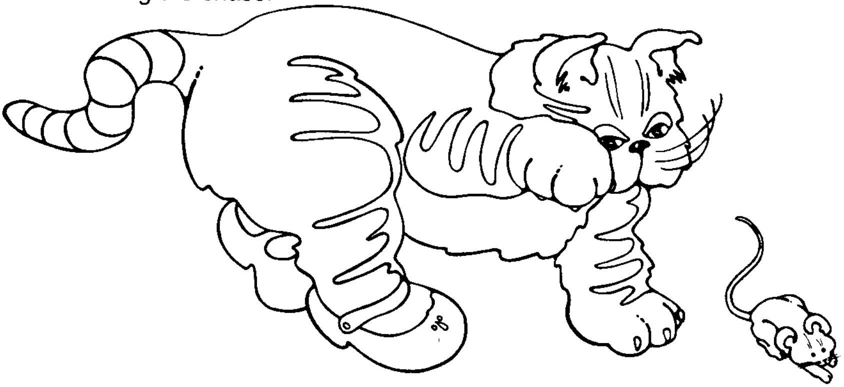
The pause:

84 Clears the screen.
85–90 Positions and prints the message.
95 Pauses while the computer counts to itself to 100.

The pounce:

When the OPTION key is pressed, the number stored at location 53279 becomes a 3.

27 If you pressed OPTION and you are over the 40th time through the loop, then "YOU GOT HIM!"
28 If not, then you missed. Pauses and goes to line 84 before resuming the chase.



A Merry Chase

To cut down on food costs, Matilda takes to catching mice. She does love to chase those mousies! One little devil keeps escaping. He just runs about like crazy. See for yourself!

```
1Ø GR.7+16
15 C.1
2Ø PL.Ø,Ø
25 FOR T=1 TO 5Ø
3Ø X=INT(16Ø*RND(1))
4Ø Y=INT(96*RND(1))
5Ø DR.X,Y
6Ø S=INT(256*RND(1))
7Ø SO.Ø,S,1Ø,8
8Ø NEXT T
1ØØ GOTO 1Ø
```

Changing colors:

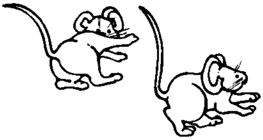
```
16 C=INT(16*RND(1))
17 SE.Ø,C,6
18 SE.4,C+5,1Ø
```

A short pause:

```
84 GR.Ø
85 POS.1Ø,1Ø
9Ø PRINT "HERE HE COMES AGAIN!!!"
95 FOR P=1 TO 1ØØ:NEXT P
```

Pounce!

```
26 REM OPTION KEY
27 IF PEEK(53279)=3 AND T>4Ø THEN PRINT "YOU GOT
   HIM!":END
28 IF PEEK(53279)=3 AND T<=4Ø THEN PRINT "MISSED!":FOR
   P=1 TO 2ØØ:NEXT P:GOTO 84
```



Main ideas: Incrementing a total
Converting cents to dollars and cents

5 T stands for the total amount of money. At the beginning of the month, the total is 0.

6 Clears the screen.

10 Prints a question.

20 Allows you to tell how many days are in the month (M).

25 Skips a line and prints the directions.

30–70 This loop will go through each day (D) in the month until it counts up to the total number of days in the month (M):

40 Prints the number of the day.

50 Allows you to tell the amount (A) that Matilda put in.

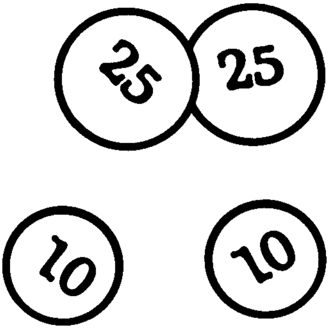
60 Adds A to the total, so you get a new total.

70 Goes back to the beginning of the loop (line 30) for the next day's deposit.

75 Skips a line.

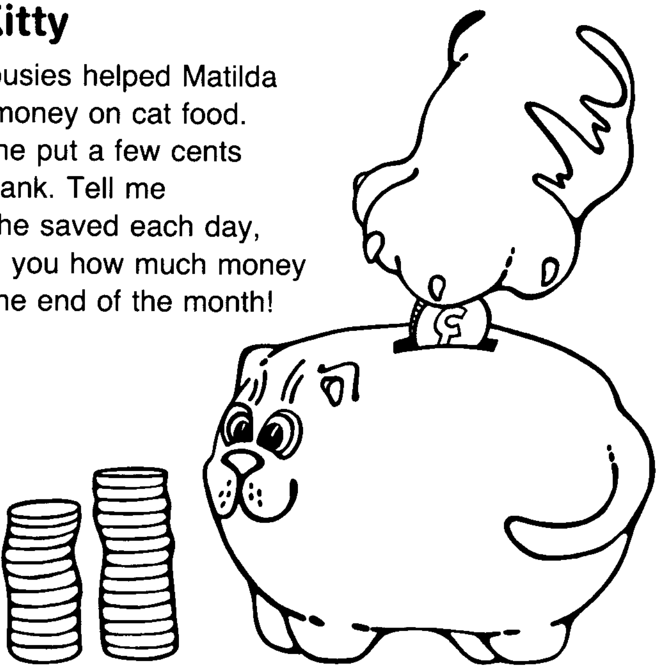
80 Prints the total amount for the month.

In this program, you should enter the cents as a whole number. For example, enter 39 for 39¢ and 104 for $1.04. The last line (line 80) of the program divides the total by 100. This makes the total come out in dollars and cents!



In the Kitty

Catching mousies helped Matilda save some money on cat food. Every day she put a few cents in her kitty bank. Tell me how much she saved each day, and I will tell you how much money she had at the end of the month!



```
5 T=Ø
6 GR.Ø
1Ø PRINT "HOW MANY DAYS IN THE MONTH"
2Ø INPUT M
25 PRINT:PRINT "NOW LIST THE CENTS FOR EACH DAY"
3Ø FOR D=1 TO M
4Ø PRINT "DAY ";D;
5Ø INPUT A
6Ø T=T+A
7Ø NEXT D
75 PRINT
8Ø PRINT "MATILDA PUT $";T/1ØØ;" IN THE KITTY"
```

Main idea: Guessing game

10 Clears the screen.

20–40 Prints the directions.

50 Selects a number from 0 to 29; then adds 20 to the number it picked. The end result is a number from 20 to 49.

60 Your chance to guess Harold's weight.

70 If your guess was too low, it prints "MORE" and goes back to line 60 so you can guess again.

80 If your guess was too high, it prints "LESS" and goes back to line 60 so you can guess again.

90 If your guess was exactly right, it goes into Graphics mode 2 and prints a big "FAT CAT!" message in the middle of the screen.

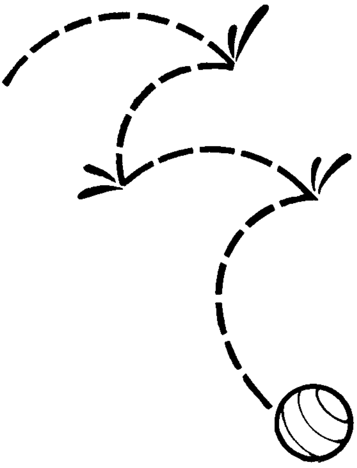
Want a different color background on the "FAT CAT!" message? If so, add this line:

```
100 SE.4,__,__
```

a number from 0 to 15 (0 is the darkest)

a number from 0 to 15 (look at the Setcolor Chart in the Appendix)

Note: Each time you run the program, you will get a different weight for Harold!



Fat Cat

In spite of the hard times, Harold grew up happy and healthy. And just guess how he weighed!

```
10 GR.0
20 PRINT "HAROLD GREW UP."
30 PRINT "GUESS HOW MUCH HE WEIGHED!"
40 PRINT "HINT:LESS THAN 50!"
50 W=INT(30*RND(1))+20
60 INPUT G
70 IF G<W THEN PRINT "MORE":GOTO 60
80 IF G>W THEN PRINT "LESS":GOTO 60
90 IF G=W THEN GR.2:POS.6,3:PRINT#6;"FAT CAT!"
```

Main ideas: Using an array for a multiple-choice test

This program uses something called an "array" to store the answers to the questions.

Answer 1 or A(1) = 2
Answer 2 or A(2) = 1
Answer 3 or A(3) = 7

(If you want to peek, the rest of the answers are in line 170!)

10 Each answer that goes on the screen (A$) is allowed 15 letters. Each question (Q$) is allowed 40 letters. There are 10 numbers stored in the answer array (A).

20 A remark.

30 Clears the screen in Graphics 1, which features larger letters than Graphics 0, but smaller letters than Graphics 2. (Look at the Graphics Sheets in the Appendix.)

40–50 Prints the directions.

60 Gets in position to start printing the list of answers.

70–100 Reads the 10 answers in line 110 and prints each one.

110 The data for the answers.

120 A remark.

130–160 Reads the 10 numbers in line 170 and stores them in the A array.

A(1)	A(2)	A(3)	A(4)	A(5)	A(6)	A(7)	A(8)	A(9)	A(10)
2	1	7	6	3	8	4	5	9	10

170 The data that goes into the A array (see lines 130 to 160).

180 A remark.

190–250 This loop prints each question, waits for a response, and checks the answer:

190 The loop will be repeated 10 times.

200 Reads a question (contained in data lines 280 and 290).

210 Prints the question in the text window.

220 Waits for you to type in the *number* of the answer.

230 Skips 2 lines. (This fixes things so that only one question at a time appears in the text window.)

240 If your response is the same as the number stored in the array, then 1 is added to the number correct.

250 Completes the loop and goes back to line 190.

260 A remark.

270 Prints your score by multiplying the number correct by 10. (Since there are 10 questions, you get 10 points for each correct answer.)

280–290 The data for the questions.

Note: It is traditional to put all data lines at the end of the program. At times, though, you might want to put them somewhere else. This program reminds you of that possibility. Remember that no matter where the lines are placed in the program, the computer reads DATA in order as it encounters READ statements.

Final Exam

Harold moved away from home. Matilda emptied her kitty bank and used the money to go to Cat College. She majored in Feline Facts. Here is part of her final exam. Can you pass it?

```
10 DIM A$(15),Q$(40),A(10)
20 REM PRINT ANSWERS ON SCREEN
30 GR.1:POS.0,3
40 PRINT#6;"PICK THE BEST ANSWER"
50 PRINT#6;"TYPE ITS NUMBER"
60 POS.0,7
70 FOR A=1 TO 10
80 READ A$
90 PRINT#6;A;" ";A$
100 NEXT A
110 DATA SIAMESE,KITTEN,MANX,CUB,
    PRIDE,PANTHER,CHEETAH,SPHINX,TIGER,MAMMALS
120 REM LOAD ARRAY OF ANSWERS
130 FOR N=1 TO 10
140 READ X
150 LET A(N)=X
160 NEXT N
170 DATA 2,1,7,6,3,8,4,5,9,10
180 REM PRESENT QUESTIONS AND CHECK ANSWERS
190 FOR N=1 TO 10
200 READ Q$
210 PRINT Q$;
220 INPUT R
230 PRINT:PRINT
240 IF R=A(N) THEN C=C+1
250 NEXT N
260 REM PRINT SCORE
270 PRINT "YOU GOT ";C*10;" %"
280 DATA BABY CAT,CAT WITH BLUE EYES,FASTEST
    CAT,BLACKEST CAT,TAIL-LESS CAT,FAMOUS EGYPTIAN CAT
290 DATA BABY LION,GROUP OF LIONS OR TIGERS,BIGGEST
    CAT,CATS BELONG TO THIS CLASS OF ANIMALS
```

Main ideas: Using control characters to create a picture
Random placement on the screen

Look at the control characters on the Control Graphics Keyboard in the Appendix. These characters are obtained by pressing the CTRL key at the same time as you press a character key. For example, to get a heart, press CTRL and the comma.

This program is hard to list on the printer because the control characters don't print. To show you how to draw these little cats, I've given you some pretty odd-looking lines. For example, look at line 18, which you should read like this: "Press CTRL–F, CTRL–G, CTRL–N, CTRL–F, CTRL–G, space, space, CTRL–V, space, CTRL–V." The end result on your video display will be something like this:

```
    18 PRINT "╱╲▁╱╲  │ │"
```

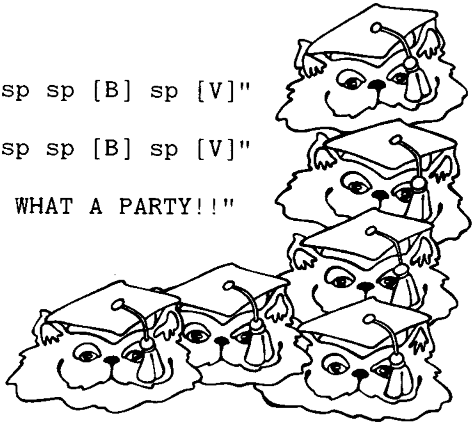
Let's see how the program works:

5 Graphics 0 clears the screen.
6–100 The loop that draws the kitty.
11 Picks a number for X from 0 to 29. (This will leave at least 10 spaces to the right for drawing the kitty.)
12 Picks a number for Y from 0 to 13. (This will leave at least 10 spaces at the bottom for drawing the kitty.)
15 Starts drawing the first line of the cat at the X,Y position picked in lines 11 and 12.
16 The directions for drawing the first line.
17 Moves down a line.
18 The directions for drawing the next line.
19 Moves down a line.
20 The directions for drawing the next line, and so forth.
105 Prints a message.
110–130 A shady ending.
110 Picks a number from 0 to 15 for the background color.
115 Starts to loop through 100 flashes.
120 SE.2 is the background color in Graphics 0. This line sets a dark shade of the color selected.
125 Sets the background in a light shade of the color selected.
130 Goes back to line 115 for the next flash.

What a Party!

It's graduation day! Matilda is so happy that she invites thirty close friends to a graduation party. Here they are!

```
5 GR.Ø:POKE 752,1
6 FOR K=1 TO 3Ø
11 X=INT(3Ø*RND(1))
12 Y=INT(14*RND(1))
15 POS. X,Y
16 PRINT "sp sp sp sp sp sp sp sp [NN]"
17 POS.X,Y+1
18 PRINT "[FGNFG] sp sp [V] sp [V]"
19 POS.X,Y+2
2Ø PRINT "[VI] sp [OB] sp sp sp [V]"
29 POS.X,Y+3
3Ø PRINT "[V] sp [P] sp [B] sp sp sp [V]"
39 POS.X,Y+4
4Ø PRINT "[GZXCF] sp sp sp [V]"
49 POS.X,Y+5
5Ø PRINT "[FMMMMMMMG]"
59 POS.X,Y+6
6Ø PRINT "[V] sp sp sp sp sp sp sp [B]"
69 POS.X,Y+7
7Ø PRINT "[GNNNNNNNF]"
79 POS.X,Y+8
8Ø PRINT "[B] sp [V] sp sp sp [B] sp [V]"
89 POS.X,Y+9
9Ø PRINT "[B] sp [V] sp sp sp [B] sp [V]"
1ØØ NEXT K
1Ø5 POS.1Ø,1Ø:PRINT "WOW WHAT A PARTY!!"
11Ø C=INT(16*RND(1))
115 FOR F=1 TO 1ØØ
12Ø SE.2,C,2
125 SE.2,C,14
13Ø NEXT F
```



Main idea: Using ASCII codes for a coded message

The ASCII codes for the common keyboard characters are listed on the opposite page and are used in this program. If you want to see all of the codes for the ATARI® Computer (ATASCII™ codes), then type in this short program:

```
10 FOR N=Ø TO 255
20 PRINT N;CHR$(N)
30 NEXT N
```

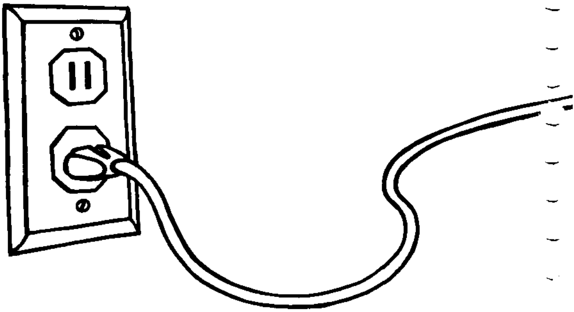
Too fast? Add the following line:

```
25 FOR P=1 TO 1ØØ:NEXT P
```

Did you remember to type "NEW"? Now for the program:

10 Clears the screen.
11 Gets rid of the cursor.
20 Positions the beginning of the message.
30 Reads the numbers in data lines 70 and 80.
40 Checks for the flag data, which signals the end of the data.
50 Prints the character related to the ASCII number.
60 Goes to line 30 to read the next number and prints its character.
70–80 The data.
100 A dummy line to freeze the image.

You may change the name of the school by using different numbers in line 70.

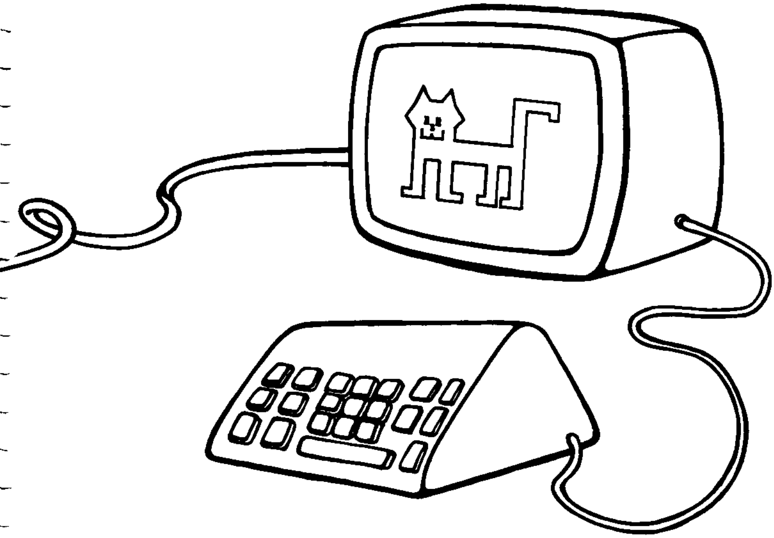


A New Career

Matilda now has a new career. She is running a computer school for young cats. Do you know what the name of her school is?
Run the program to find out!!!

```
10 GR.0
11 POKE 752,1
20 POS.10,10
30 READ N
40 IF N=-1 THEN 100
50 PRINT CHR$(N);
60 GOTO 30
70 DATA 67,79,77,80,85,45,75,73,84,84,73,69,83,33
80 DATA -1
100 GOTO 100
```

32	
33	!
34	"
35	#
36	$
37	%
38	&
39	'
4Ø	(
41	)
42	*
43	+
44	,
45	-
46	.
47	/
48	Ø
49	1
5Ø	2
51	3
52	4
53	5
54	6
55	7
56	8
57	9
58	:
59	;
6Ø	<
61	=
62	>
63	?
64	@
65	A
66	B
67	C
68	D
69	E
7Ø	F
71	G
72	H
73	I
74	J
75	K
76	L
77	M
78	N
79	O
8Ø	P
81	Q
82	R
83	S
84	T
85	U
86	V
87	W
88	X
89	Y
9Ø	Z



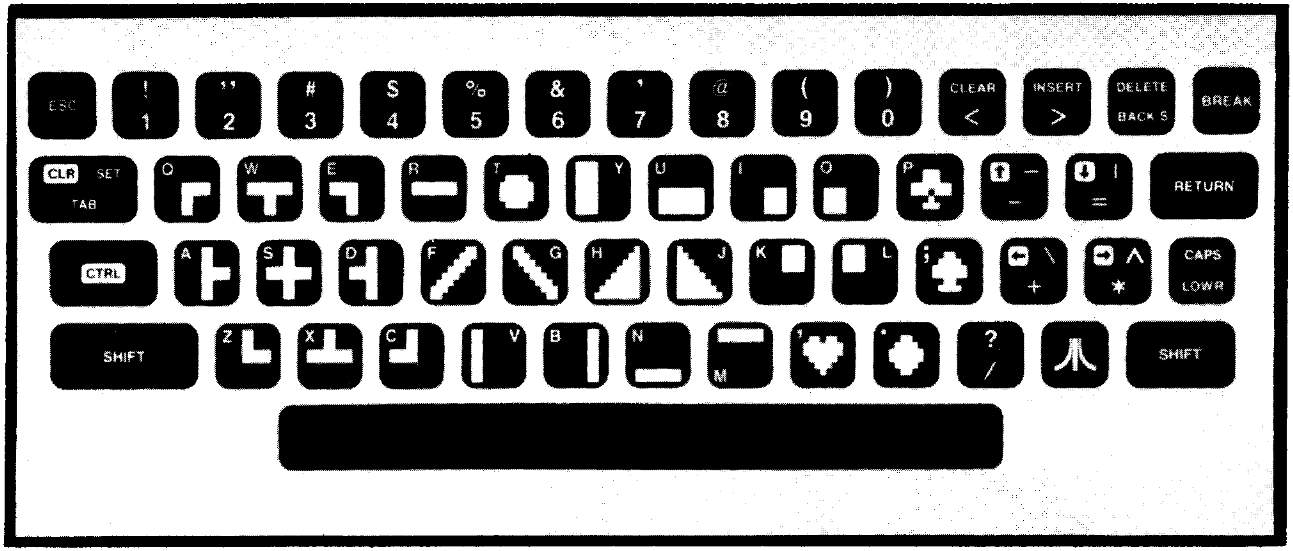
APPENDIX

Abbreviations

This is a partial list of the abbreviations available. Use them, but remember what they stand for!

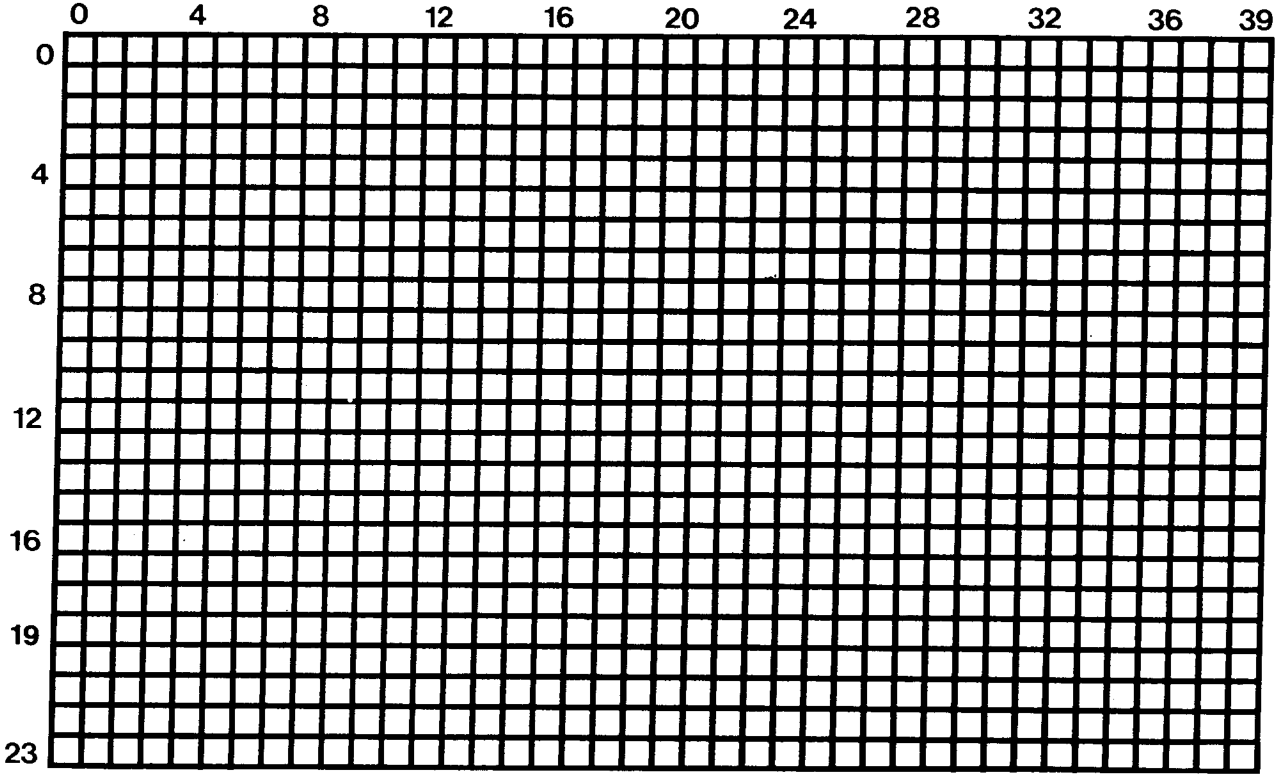
C.	COLOR
D.	DATA
DR.	DRAWTO
F.	FOR
G.	GOTO
GOS.	GOSUB
GR.	GRAPHICS
I.	INPUT
L.	LIST
LOC.	LOCATE
N.	NEXT
PL.	PLOT
POS.	POSITION
PR.	PRINT
?	PRINT
RES.	RESTORE
RET.	RETURN
SE.	SETCOLOR
SO.	SOUND

Control Graphics Keyboard

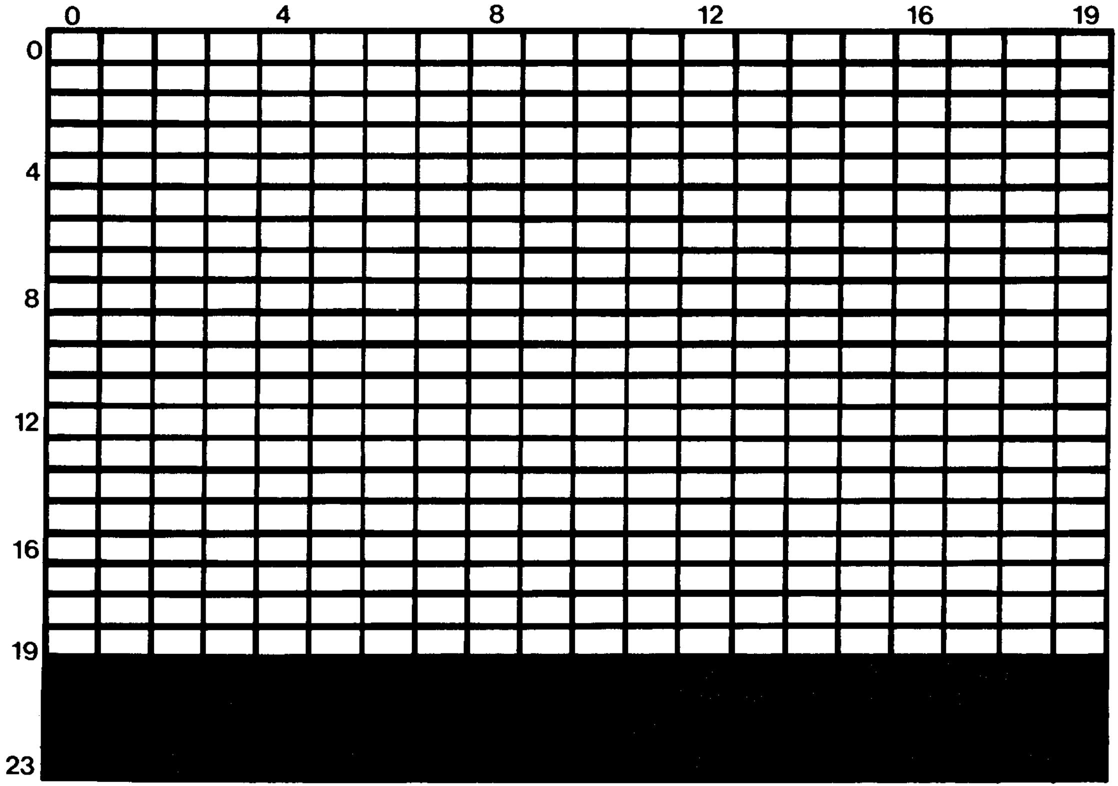


Photograph reprinted by permission of Atari, Inc. © 1980

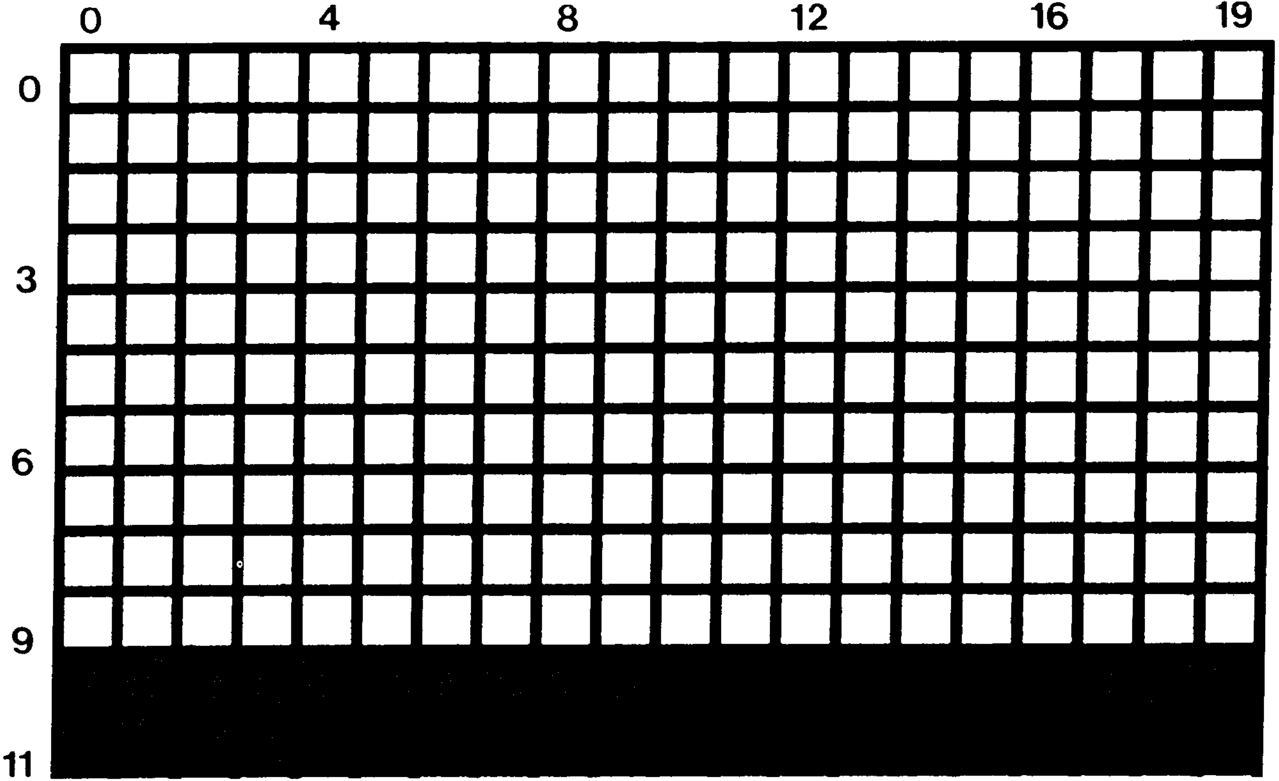
Graphics 0 Sheet



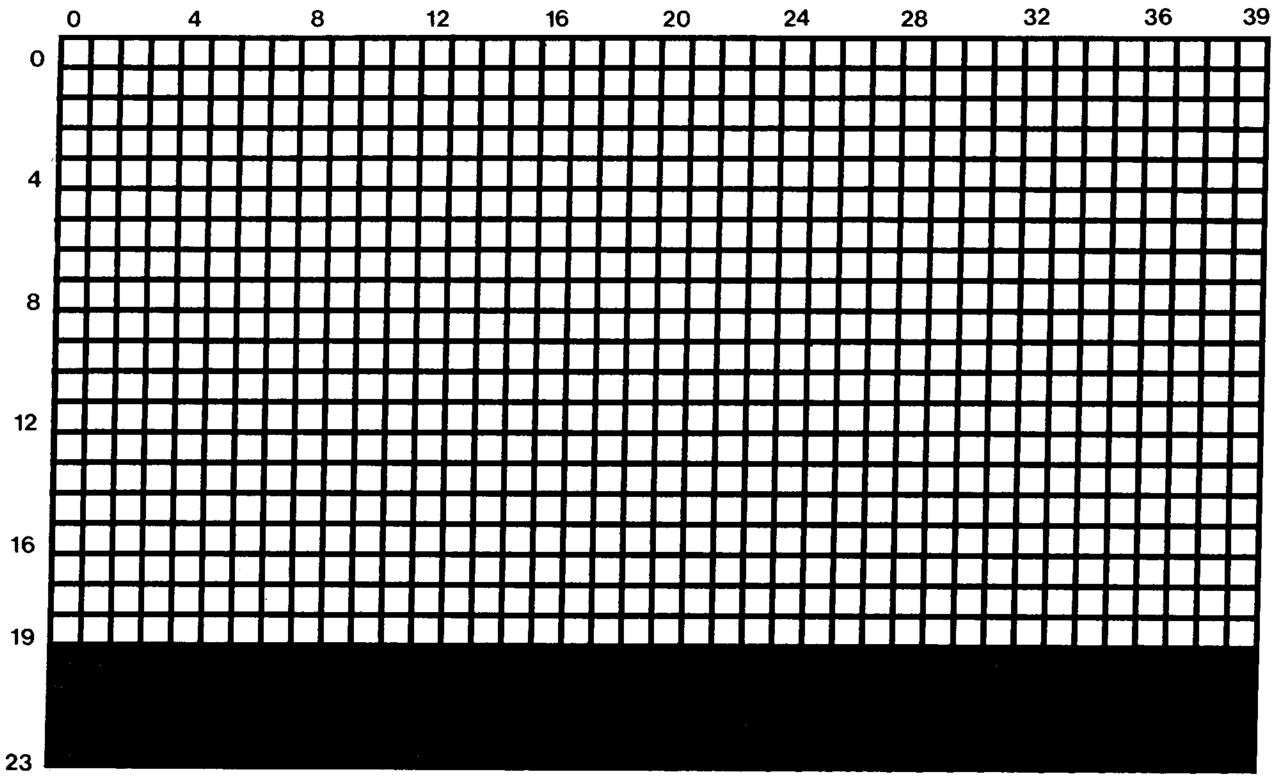
Graphics 1 Sheet



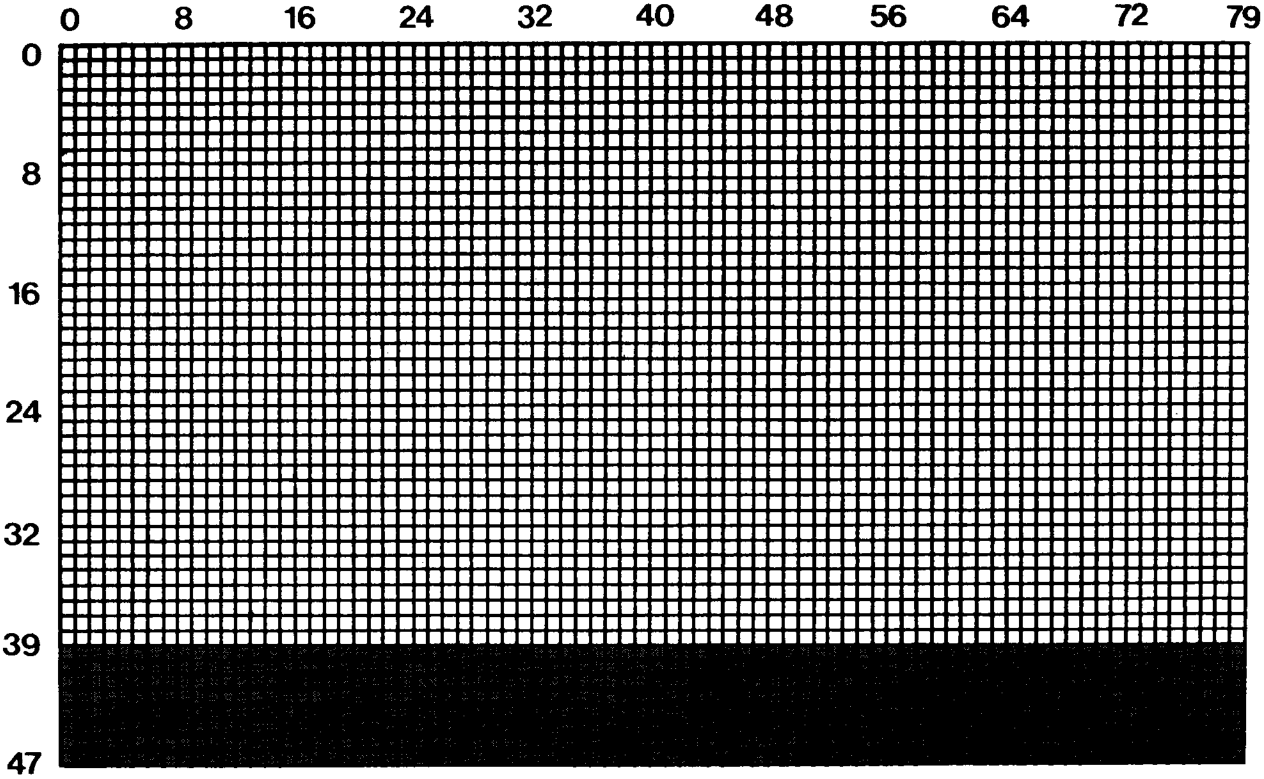
Graphics 2 Sheet



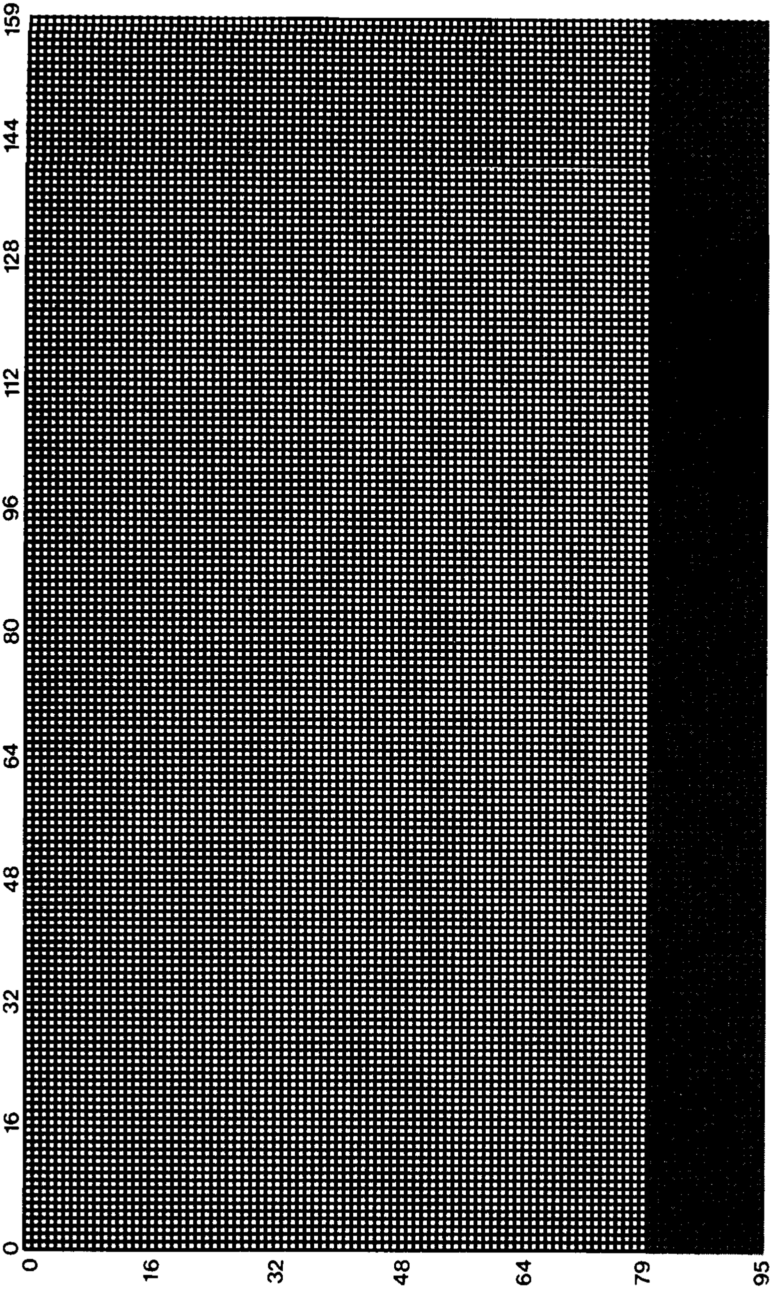
Graphics 3 Sheet



Graphics 5 Sheet

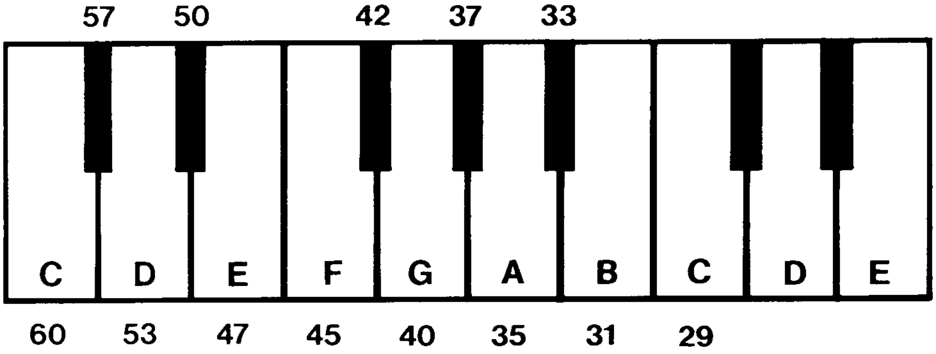


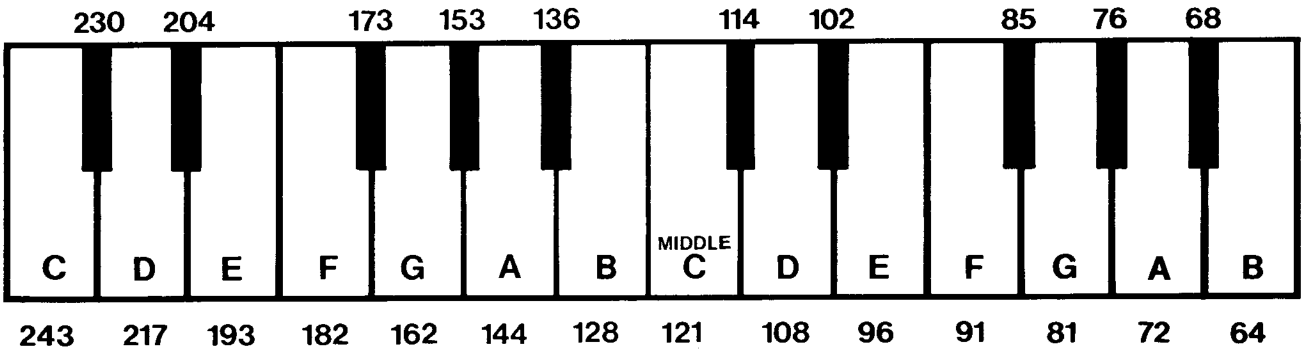
Graphics 7 Sheet



Musical Notes

Use the numbers assigned to the notes to create your own computer songs!





Saving Programs on Cassette

Are you having problems? I find this procedure to be more reliable than the usual CSAVE–CLOAD. It takes a little longer, but it saves a lot of time and aggravation in the long run.

1. Position the tape at a new location. (Don't tape over old programs!) Remember the number.
2. Type LIST "C:" and press RETURN.
3. The computer will buzz twice. Make sure two buttons on the cassette recorder are down, PLAY and RECORD. Then press RETURN again.
4. When the program is saved, the screen will say READY.
5. To verify that your program is saved, reposition the tape and type ENTER "C:".
6. The computer will buzz once. Make sure that the PLAY button is down. Press RETURN.
7. If you get no errors, the program is properly saved. If you do get an error message, panic not! The original program is still in memory. Find a new place on the tape and try to save it again, beginning with step 2.
8. Save your money for a disk drive.

Setcolor Chart

Color it in to help you out!

0		black
1		rust
2		red-orange
3		dark orange
4		red
5		dark lavender
6		cobalt blue
7		ultramarine blue
8		medium blue
9		dark blue
10		blue-gray
11		olive green
12		medium green
13		dark green
14		orange-green
15		orange

$6.95

CLAIRE BAILEY PASSANTINO

MATILDA THE COMPUTER CAT

Matilda the Computer Cat is delightful! Your child will immediately fall in love with Matilda and enjoy the various predicaments and adventures—everything from lost kitten mittens to chasing mice for dinner—that take place on the computer screen.

Claire Bailey Passantino has once more created an environment that encourages your child to explore the magic of your home computer. Matilda's story unfolds through bit-sized programs and activities designed to excite young eyes, ears and minds. Each program is accompanied by programming hints and suggestions for parents and teachers.

Help your child learn, grow and play—the computer way—with Matilda.

Some of the activities in Matilda include:
Meet Matilda ★ Our Song ★ A New Rug! ★ Fleas! ★ The Cat's Pajamas ★ Lost Mittens ★ Cat Heritage ★ A Merry Chase ★ Final Exam ★ A New Career

Other Itty Bitty Bytes Books:
Itty Bitty Bytes of Space
School Days

For more information on Creative Pastimes Books write to:

Reston Computer Group
Reston Publishing Company, Inc.
11480 Sunset Hills Road
Reston, Virginia 22090

0 21898 43233 9

0-8359-4323-2